
GitHub and Vanilla Forums Integration Guide

Vanilla + GitHub

Introduction

If you are using Vanilla as a support forum for your software product, this integration allows you to create issues in a GitHub repository from comments posted in Vanilla.

This guide outlines how to setup the integration and how it works.

Note: This integration is currently in beta

1. Turning On The GitHub Plugin in Vanilla

- Only Administrators can enable plugins in Vanilla. From the Addons page of the Dashboard (Dashboard > Addons > GitHub), enable the GitHub plugin.

This plugin will only be visible to customers subscribed to the Corporate, Enterprise or VIP plan. If you do not see the plugin, please contact your account manager for more information.

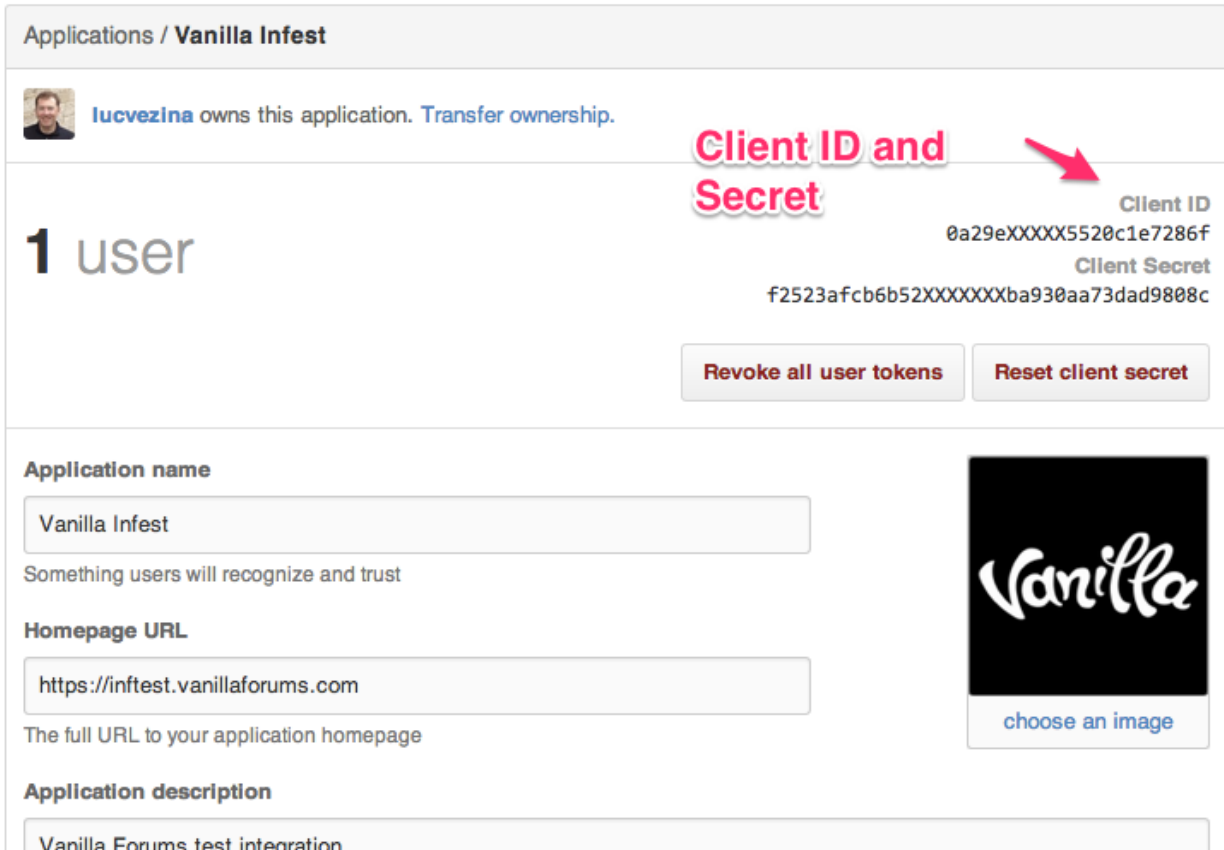
2. Configuring GitHub

Configuration of GitHub consists in creating a new application. This will allow users to authenticate their Vanilla account to their GitHub account and create issues. Authentication is done using oAuth and GitHub passwords will not be passed to Vanilla or stored in Vanilla.

2.1 Create a new application in GitHub

- Go to <https://github.com/settings/applications/new>
- Enter the following information:
 - Application Name: **Vanilla**
 - Homepage URL: **Your forum URL**
 - Logo Image: Save the following image to your desktop and upload it:
<https://vanillaforums.com/applications/vfmarketing/design/images/125-vanilla-logo.png>
 - Application Description: **Integration of Vanilla Forum to GitHub to Create Issues from forum comments**
 - Authorization callback URL:
https://[FORUM-URL]/profile/githubconnect, where [FORUM] is the name you gave your vanilla forum when it was created or the name of your custom domain. The call back URL must be **HTTPS**. The URL can be copied from the instructions on the plugin settings in the Vanilla Dashboard.
 - Click 'Register Application'

- Obtain the Client ID and Client Secret. These values will be entered into the Vanilla to connect the applications later on.



The screenshot shows the GitHub Applications page for an application named "Vanilla Infest". At the top, it says "Applications / Vanilla Infest". Below that, it shows the owner's profile picture and name "lucvezina" with a link to "Transfer ownership". The main content area displays "1 user" on the left and the "Client ID and Secret" on the right. The Client ID is "0a29eXXXXX5520c1e7286f" and the Client Secret is "f2523afcb6b52XXXXXXba930aa73dad9808c". There are two buttons: "Revoke all user tokens" and "Reset client secret". Below this, there are form fields for "Application name" (Vanilla Infest), "Homepage URL" (https://infest.vanillaforums.com), and "Application description" (Vanilla Forums test integration). On the right side, there is a "choose an image" button and a preview of the application logo, which is the Vanilla logo.

3. Configure the Integration Settings in Vanilla

Configuring the plugin will allow Vanilla to authenticate to GitHub and specify the GitHub repositories where Issues will be created.

3.1 Client ID and Client Secret

- Go to the GitHub Settings page (Dashboard > Forum > GitHub or Dashboard > Addons > GitHub > Settings)
- Enter the Client ID and Client Secret that was obtained in step 2.1

3.2 Repositories

- Enter the Repositories where Issues can be created. Use the format 'username/reponame'. Several repositories can be added, each on a separate line.

Repositories

List of Repositories separated by newline. Enter the GitHub username and repo, E.g. 'username/reponame'

```
John0x00/githubdemo  
John0x00/forumbugs  
John0x00/frontend
```

3.3 Setting Permissions in Vanilla

- Only users that are in a Role that have the 'Staff' permission are able to create GitHub Issues. The Staff permission may not be a default on Roles such as Administrator or Moderator. Please check the Role permissions for users that should have permission to create Issues.

3.4 Global Login Option

There are two ways of authenticating Vanilla to GitHub. The recommended way is for each staff member to authenticate themselves using their own GitHub credentials. This way, you will be able to see which staff member created the Issue in GitHub.

The other way is to use a Global Login. This allows staff members that do not have a GitHub license to create Issues.

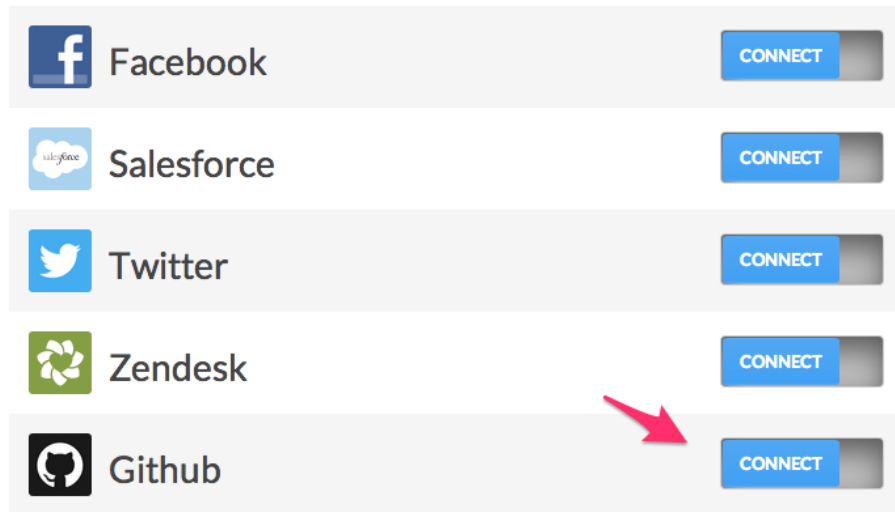
Authentication of Vanilla users to GitHub is done using OAuth. GitHub usernames and passwords are never entered into Vanilla and are not stored in Vanilla. The authentication uses secure tokens that can be revoked at any time by your GitHub Administrator.

- To configure the Global Connection, click Enable. You will be redirected to GitHub.com and asked to enter username and password for the GitHub account that you want to use to make the global connection.

4. Using the GitHub Integration

4.1 Authenticating Individual Users

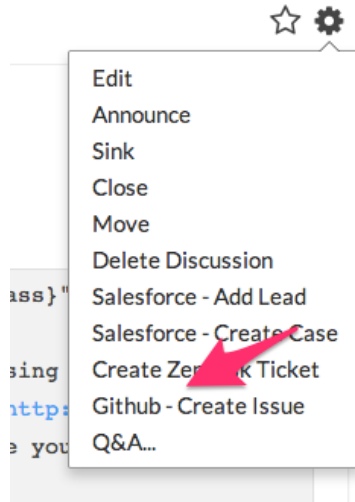
- Individual staff members must authenticate themselves to GitHub before being able to create Issues. (This is not necessary if you are using the Global Login option.) To authenticate yourself, go to Edit Profile > Social Logins and connect to GitHub.



- If a staff members attempts to create an Issues and is not authenticated, a pop-up will prompt the user to do so.

4.2 Creating an Issue

- To create an Issue, select 'GitHub - Create Issue' from the gear icon drop down.



- Name and email will be pre-populated. Other required fields will need to be filled out.

A screenshot of the 'Github Create Issue' form. The form is titled 'Github Create Issue' and has a close button (x) in the top right corner. It contains the following fields:

- Repository:** A dropdown menu showing 'John0x00/githubdemo'.
- Title:** A text input field containing 'What's wrong with this code?'.
- Body:** A text area containing the text: 'It's giving me an error: <body id="{ \$BodyID}" class="{ \$BodyClass}"> <!--[if !IE 8]> <p class="browsehappy">You are using an outdated browser. Please upgrade your browser to improve your experience.</p>'. Below the text area is a blue button labeled 'CREATE ISSUE'.

- The Issue will be created in GitHub and an information panel will be displayed (to staff only) under the post. The Lead status will be updated each time the page is loaded.

The screenshot shows a light blue information panel for a GitHub issue. It features the GitHub logo and the text "Issue · Github" followed by "5:57PM by agent007". Below this, there are two rows of data: "Issue Nu..." with the value "7" and "Last Upda..." with the value "5:57PM". To the right, the "State" is listed as "open".

- Issues are then displayed in GitHub

What's wrong with this code? #1

Open lucvezina opened this issue 12 days ago · 0 comments

The screenshot shows a comment from user "lucvezina" made 12 days ago. The comment text reads: "What's wrong wit this..." followed by "This Issue was generated from your [forums](#)". The comment box includes a profile picture of the user, a timestamp, and an "Owner" label with an edit icon.

4.3 Notes:

- If your community users are posting snippets of code and you have enable the Advanced Editor in Vanilla, we recommend setting the editor in Vanilla to Markdown so that the code formatting carries over to GitHub.